(12) UK Patent Application (19) GB (11) 2 280 524 (13) A

(43) Date of A Publication 01.02.1995

(21) Application No 9411924.5

(22) Date of Filing 14.06.1994

(30) Priority Data
(31) 08082051 (32) 24.06.1993 (33) US

(51) INT CL[6]
G11C 7/00 , G06F 12/06

(52) UK CL (Edition N)
G4C C700B
G4A AMX

(56) Documents Cited
EP 0149451 A2

(58) Field of Search
UK CL (Edition M) G4A AMX , G4C C700B C800M
INT CL[5] G06F 12/06 , G11C 7/00 8/00
Online databases: WPI

(71) Applicant(s)
PiE Design Systems Inc

(Incorporated in USA - California)

440 Clyde Avenue, Mountain View, California 94043, United States of America

(72) Inventor(s)
Thomas B Huang

(74) Agent and/or Address for Service
J A Kemp & Co
14 South Square, Gray's Inn, LONDON, WC1R 5LX, United Kingdom

(54) Method and apparatus for configuring memory circuits

(57) A memory circuit definition is implemented in static memory circuits having a plurality of access ports, such as three bidirectional access ports (206, Fig. 3), wherein each port is configured for read or write access. The memory circuit definition defines initial contents, depth, width, and bank selection in the memory circuits according to predefined configuration values, as well as, for each access port, whether that access port is configured for read or write. Port access occurs during time slots, which are based on external clock signals and memory circuit access times. Different memory circuit definitions may be implemented such that access ports are accordingly reconfigured. The memory 100 is described in more detail, (Fig. 2) showing a memory array (200) and related circuitry.

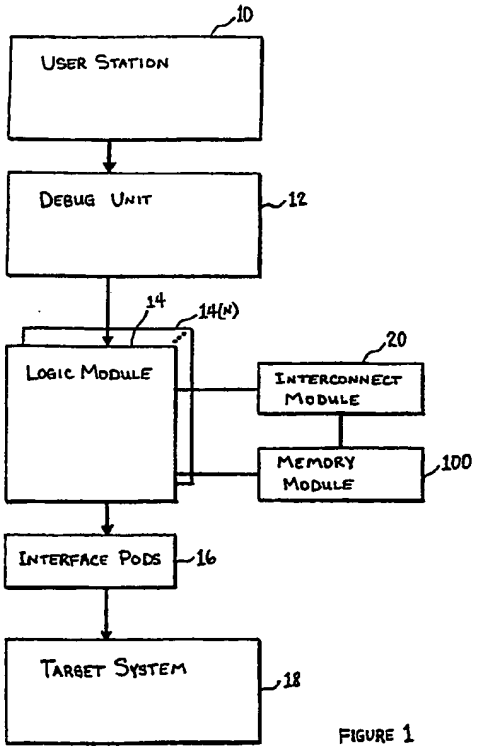


FIGURE 1

GB 2 280 524 A

At least one drawing originally filed was informal and the print reproduced here is taken from a later filed formal copy.

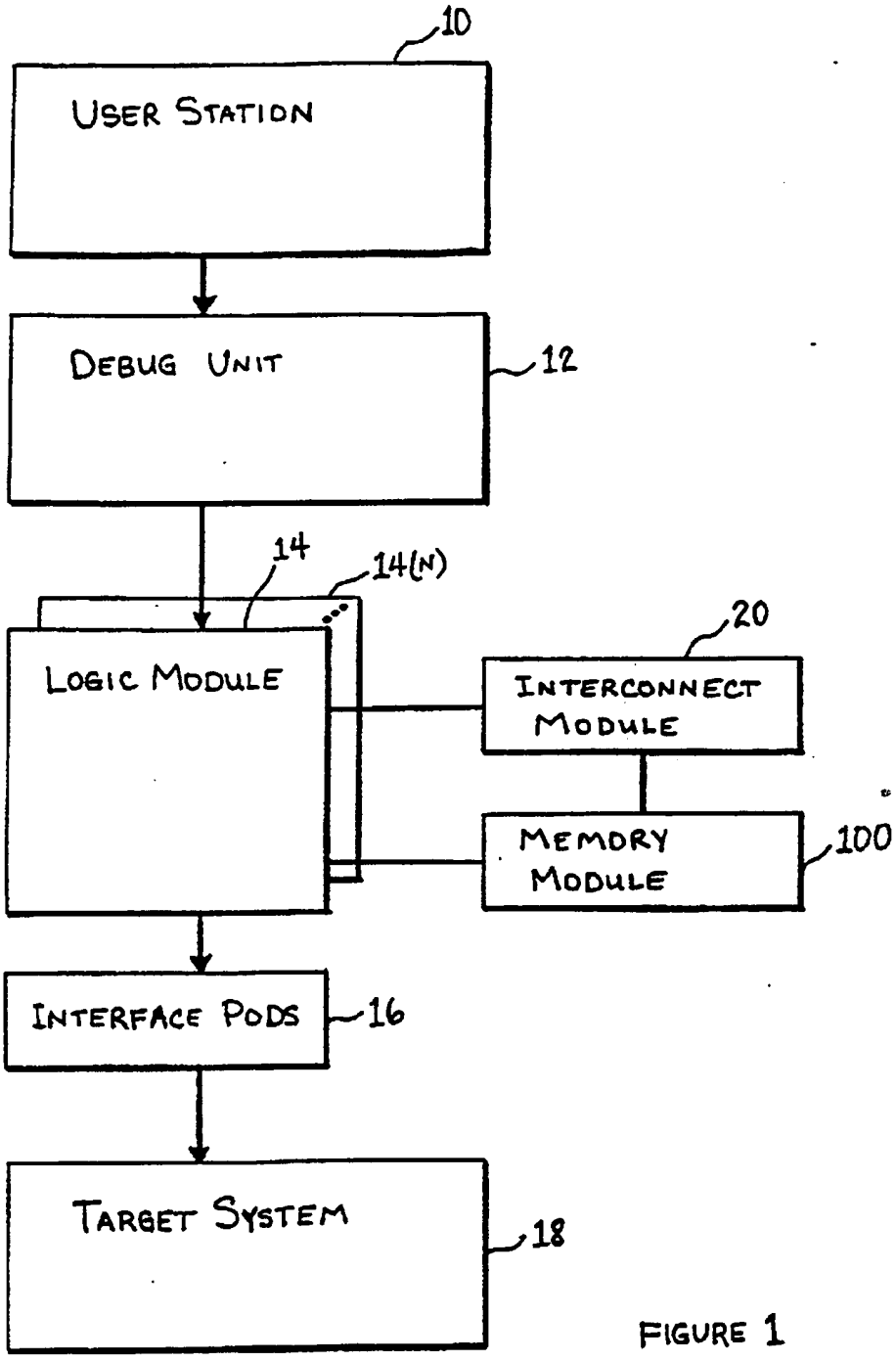

10
USER STATION
DEBUG UNIT
12
14
14(N)
LOGIC MODULE
20
INTERCONNECT MODULE
MEMORY MODULE
100
INTERFACE PODS
16
TARGET SYSTEM
18

FIGURE 1

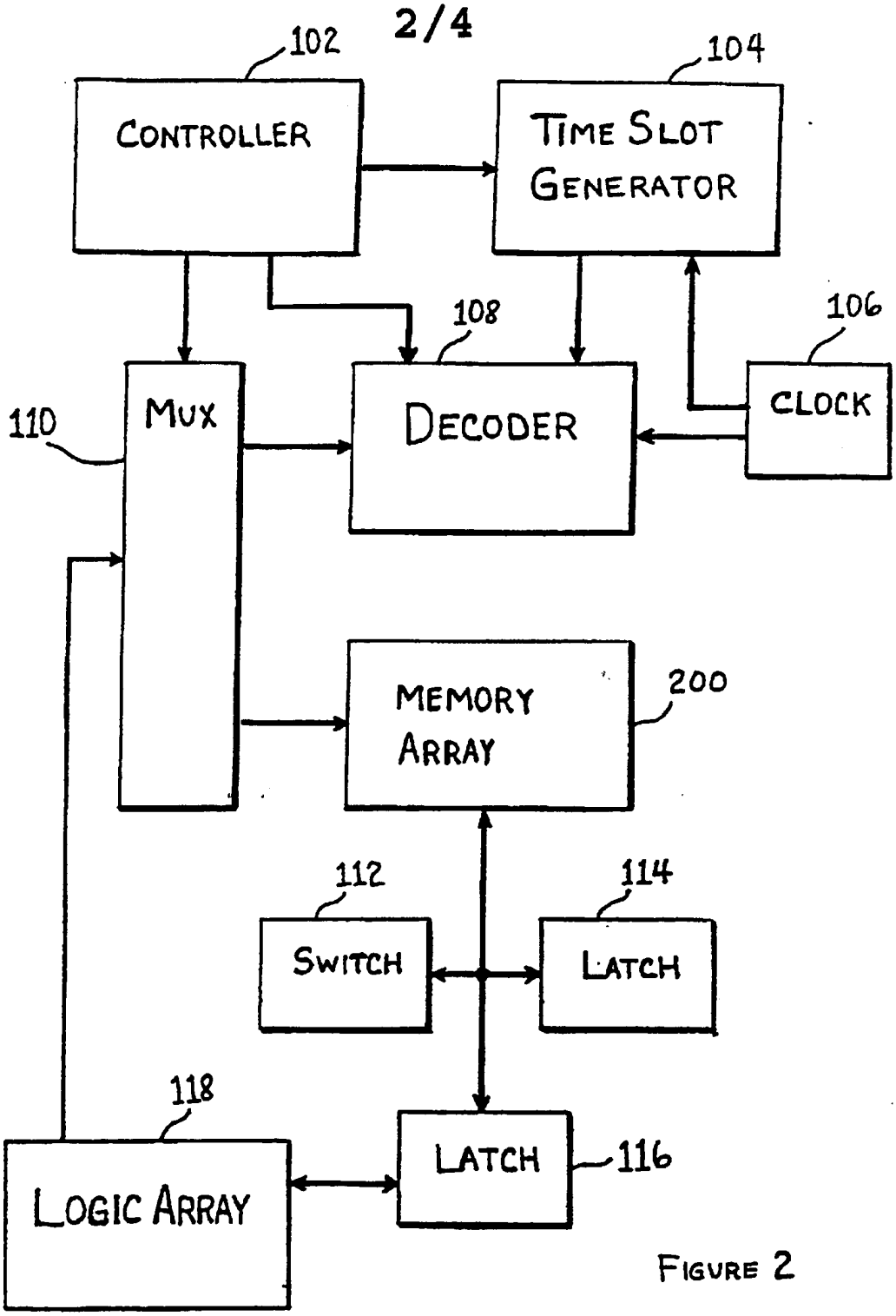
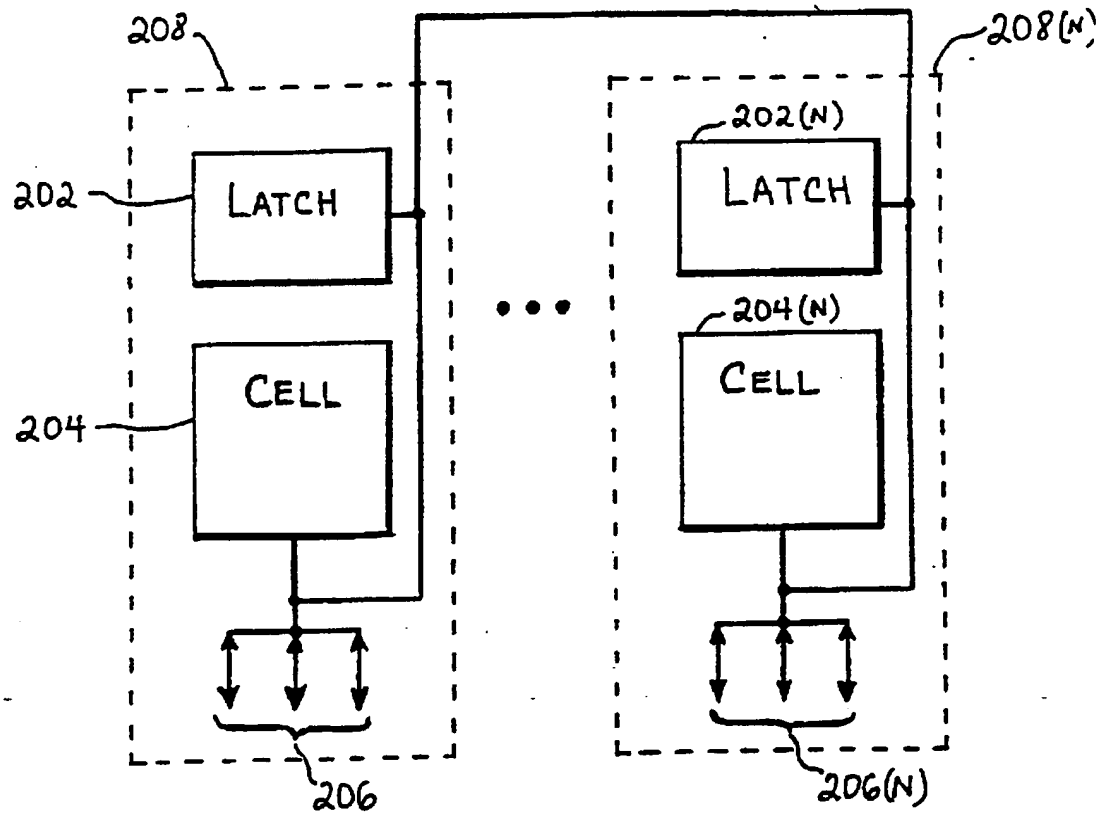

102
CONTROLLER
104
TIME SLOT GENERATOR
108
106
110
MUX
DECODER
CLOCK
MEMORY ARRAY
200
112
114
SWITCH
LATCH
118
LOGIC ARRAY
LATCH
116

FIGURE 2

100



200

FIGURE 3

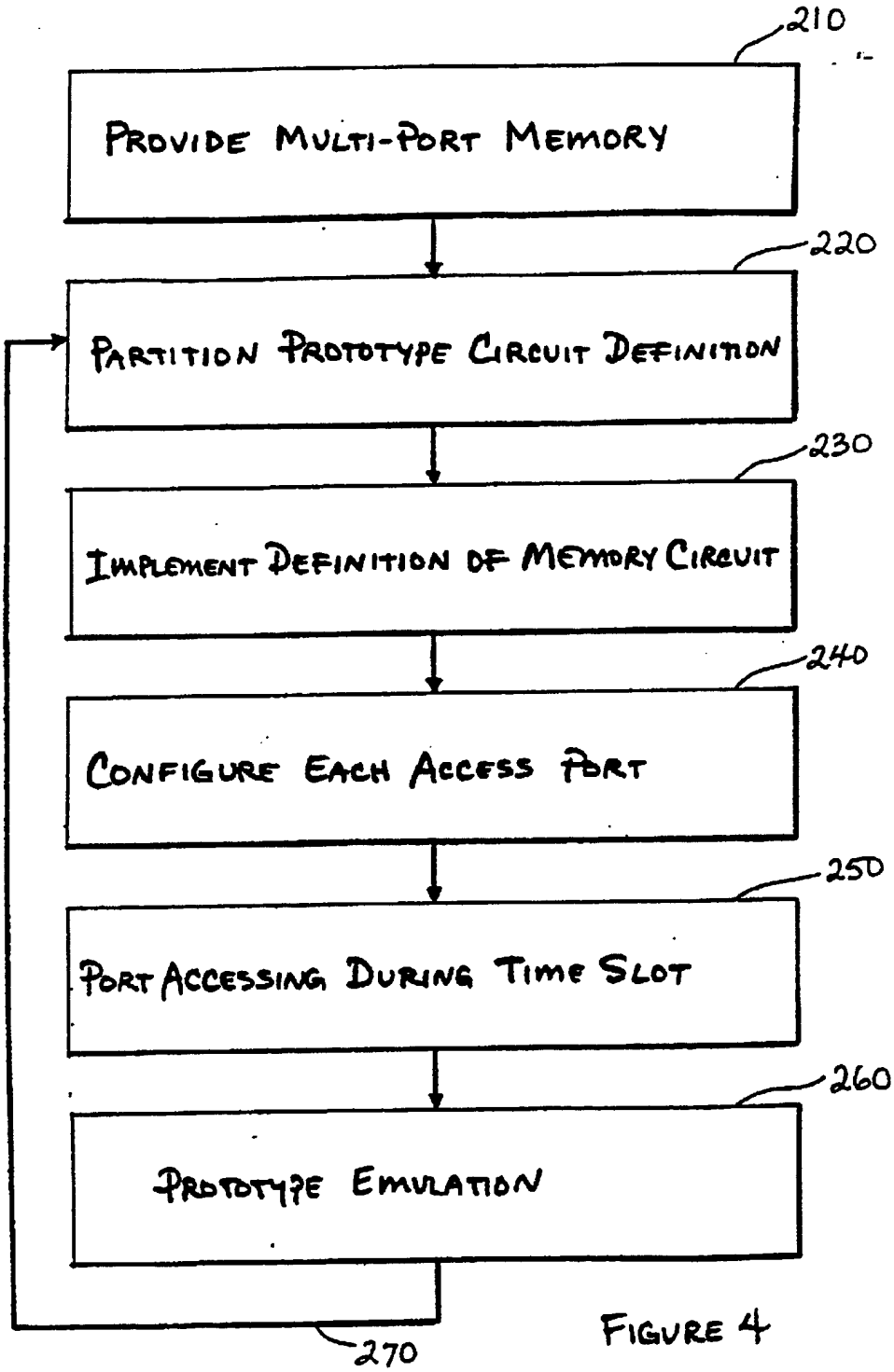

210
PROVIDE MULTI-PORT MEMORY
220
PARTITION PROTOTYPE CIRCUIT DEFINITION
230
IMPLEMENT DEFINITION OF MEMORY CIRCUIT
240
CONFIGURE EACH ACCESS PORT
250
PORT ACCESSING DURING TIME SLOT
260
PROTOTYPE EMULATION
270

FIGURE 4

METHOD AND APPARATUS FOR CONFIGURING MEMORY CIRCUITS

This invention relates to configurable electronic storage devices, particularly to memory cells configured for emulating circuits including digital memory.

In the area of electronic design automation (EDA), various software and associated hardware are used to define and verify circuit designs. Particularly to achieve more accurate verification of circuit designs, EDA tools, known as "emulators," are employed to construct functional representations of prototype circuit definitions. Because such emulated representations allow a circuit designer flexibly to operate or develop a target system coupled to the emulated representation, even before the prototype circuit or hardware is actually manufactured, overall design time and cost is reduced significantly.

Although conventional emulation tools are suited for defining and verifying logic circuits, such tools are not used easily for emulating memory circuits, particularly multi-ported memory circuits configured in various storage configurations. It would be desirable, therefore, to provide memory circuits which are more easily configurable to emulate prototype circuit designs.

The invention resides in implementing a memory circuit definition in at least one of various memory circuits having multiple access ports, wherein each access port of each memory circuit, in which the memory circuit definition is implemented, is configured for either reading from or writing to each such memory circuit.

Preferably, each memory circuit comprises static memory cells having three bidirectional access ports, and the memory circuit definition defines initial storage contents, depth, width, and bank selection in the memory circuits according to predefined configuration values. Also the memory circuit definition may define, for each access port of memory circuits having the implemented memory circuit definition, whether such access port is configured for reading or writing.

Optionally, the configured access ports may be accessed during predefined time slots to read from or write to the memory circuits. Separate time slots may be predefined for reading from or writing to the memory circuits, as opposed to non-memory logic circuits. Time slots may be predefined as a function of external clock signals and memory circuit access times.

Optionally, a prototype circuit definition may be partitioned into a logic circuit definition for implementation in a reconfigurable logic module and into the memory circuit definition for implementation in at least one of the memory circuits. The reconfigurable logic module may be interconnected directly, or through a programmable interconnect module, to the memory circuits.

To emulate the partitioned prototype circuit definition, the reconfigurable logic module and the memory circuits are interconnected and coupled to a target system, which is then operated after so coupled. During emulation, the target system may access the access ports to read from or write to the memory circuits. The target system may also cause a different memory circuit definition to be implemented in the memory circuits, whereupon the access ports would be reconfigured accordingly.

The invention will be further described by way of non-limitative example, with reference to the accompanying drawings, in which:-

FIG. 1 is a block diagram of an electronic design automation (EDA) system showing user station 10, debug unit 12, and logic modules 14 coupled to interconnect and memory modules 20, 100 and to target system 18 through interface pods 16.

FIG. 2 is a block diagram of memory module 100 showing memory array 200 and related circuitry.

FIG. 3 is a block diagram of memory array 200 showing memory circuit 208 including cells 204 and latches 202.

FIG. 4 is a flow chart showing generally a method embodying the present invention.

FIG. 1 is a block diagram of an electronic design automation (EDA) system showing engineering workstation or user station 10 coupled to debugger or debug unit 12, logic block modules (LBM) 14, programmable interconnect module (PIM) 20, configurable memory module 100, interface pods 16, and target system 18. The EDA system, which is configured for emulation similarly to the "MARS" logic emulation systems designed by PiE Design Systems, Inc. (Sunnyvale, CA), is used by a circuit or system designer to define and verify circuit or system functionality using bread-boarding or emulation techniques. Additional LBMs 14(N) are used to emulate more complex circuits or systems.

User station 10 includes a conventional processor which may couple to a network of similar processors for communication or distributed processing. User station 10 contains computer-aided engineering and design (CAE/CAD) software EDA tools for defining and verifying prototype circuit designs. For example, a circuit designer may use such tools to enter circuit schematics or synthesize logic gates to define or design circuit or system functionality or representation, and then use such tools to simulate or verify the defined functionality.

The circuit designer may perform additional verification by emulating the defined functionality. Emulation enables the circuit designer actually to operate an emulated representation of the prototype circuit in target system 18, in which the prototype circuit is intended to operate.

Initially, before a prototype circuit may be emulated, the circuit designer causes the EDA tools to define, and preferably verify, a particular circuit design. Once the circuit design is defined or verified, the circuit designer causes the EDA tools to generate a conventional circuit file, or netlist, which provides a textual listing (i.e., in "ASCII" format) of components and interconnections within the circuit design. In accordance with the present invention, each circuit design is defined to include at least one memory component as well as logic components.

Debug unit 12, which is coupled to user station 10, allows the circuit designer to transfer or down-load the generated netlist from user station 10 to logic modules 14, interconnect module 20, and memory module 100. Debug unit 12 also serves as a functional tester and logic analyzer for verifying the operation of the emulated representation of the prototype circuit design. Debug unit 12 may operate in either functional test (debug) mode or emulation (run) mode.

After a particular netlist is generated, the circuit designer may cause the EDA tools to receive the netlist to cause the circuit design or definition contained therein to be implemented physically or "down-loaded" into various reconfigurable logic circuits or field-programmable gate arrays (FPGAs) included in logic modules 14, in the case of logic components, and into memory cells or static random access memory (SRAM) included in memory module 100, in the case of memory components.

Emulation is achieved by programmably configuring, according to the netlist, the actual interconnection and functional implementation of reconfigurable logic circuits included in logic modules 14, and, as necessary, the actual interconnection of reconfigurable electrical paths in interconnect module and the actual memory configuration and contents in memory module 100.

The circuit designer invokes a design-import command to cause the EDA tools to check, among other things, whether all logic and memory components included in the netlist are available or stored in a predefined library or database of logic and memory representations, as provided respectively in logic module 14 and memory module 100. Preferably, if a particular logic or memory component is not available in the library, then the EDA tool may generate a particular configuration of a corresponding generic logic or memory

component. In this way, using the EDA tool to generate or construct a particular component configuration, the circuit designer may customize or specify more completely certain functionality or operational parameters of such particular configuration.

After the netlist is imported, the circuit designer may invoke a partitioner program included in the EDA tools to determine which components included in the netlist are logic and memory components. Preferably, components which include binary or digital information configured or arranged for temporary or permanent storage in conventional or various combinatorial, sequential, or state devices are determined to be memory circuits or components. Examples of memory components include core memory arrays having single or multiple ports, cache arrays (including store, tag and status arrays,) multi-ported register files, microcode read-only memory (ROM), on-chip random-access memory (RAM), first-in/first-out (FIFO), and processor memory (including instruction and data caches, general purpose registers, and local and main storage RAM).

Other components included in the netlist, typically including conventional combinatorial, sequential, or state circuits configured or designed for non-storage applications, are determined to be logic circuits or components. It is possible, as designated by the design engineer, to employ

conventional logic circuits for storage memory-type design applications.

After determining which netlist components are logic components and which are memory components, the partitioner program partitions the netlist for down-loading or actual functional implementation of the components and interconnections to logic modules 14, interconnect module 20 or memory module 100.

Logic components are down-loaded to logic modules 14 to configure logic circuit portions and associated interconnections, preferably using conventional techniques for programming or configuring FPGAs or configurable logic blocks (CLBs) contained therein according to the partitioned netlist. Memory components are down-loaded to memory module 100 as described further herein. Optionally, memory components may be down-loaded to logic modules 14 for equivalent functional implementation therein.

Interconnections in the netlist may be down-loaded either to logic module 14, using available configurable interconnect on each logic module 14 or FPGAs contained therein, or to interconnect module 20, using programmable interconnect paths or configurable cross-bar switches available therein.

The entire prototype circuit design or definition is implemented functionally and physically for emulation when all or the desired portions or sub-netlists of logic and memory components and interconnections in the netlist are down-loaded to logic modules 14, memory module 100 and, optionally, interconnect module 20. Upon such configuration, the prototype circuit design implemented in logic modules 14, interconnect module 20, and memory module 100 may be coupled through various interface pods or signal connections 16 to a conventional port or signal socket in target system 18 for actual functional operation or emulation therein.

During emulation, target system 18 operates under normal or close-to-normal conditions or timing, by applying thereto appropriate data, control, test, power, ground and other appropriate signals, vectors or stimuli. In this way, the circuit designer may debug or test the functionality of target system 18 or the prototype design temporarily configured for emulation in logic modules 14, interconnect module 20 and memory module 100.

Additionally, the circuit designer may determine that various reconfiguration or functional variation to the interconnections or components included in the netlist are necessary to modify or change the prototype circuit design. Such modification is achieved by the circuit designer causing the EDA tools to redefine, and if necessary, reverify, the

prototype circuit design. Virtually unlimited number and extent of different or similar modifications are possible because the functional implementation of logic circuits and interconnections in logic modules 14, memory circuits in memory module 100, and interconnections in memory module 100 are reconfigurable or reprogrammable using the EDA tools.

FIG. 2 is a block diagram of memory module 100 showing memory array 200 and related circuitry, including "JTAG" processor or controller 102, time slot or signal generator 104, memory configuration decoder 108, external clock or oscillator 106, multiplexer (MUX) 110, reconfigurable logic or "beta-alpha" array 118, and bidirectional switch or latches 112, 114, 116.

In FIG. 3, memory array 200 is shown with various interconnected memory circuits 208 including digital storage circuits or cells 204, data latches or switches 202, and configurable, bidirectional read/write ports 206. Preferably, cell 204 includes eight 3-port static random access memory (RAM), wherein each cell 204 is a 32K X 32 SRAM having 25-ns access time. The access direction for each port 206 is configured or programmed as either read mode or write mode for appropriate access to corresponding cell 204, preferably by configuring or programming the signal switching direction of switch 112, latch 114 or latch 116 coupled thereto.

Controller 102 receives partitioned netlist information including logic and memory components, configuration and initialization state information and interconection, representing prototype circuit designs, which are down-loaded by the EDA tools from user station 10 or debug unit 12. Logic component or associated interconnection, configuration, or initialization information are down-loaded to program or configure reconfigurable logic devices or FPGAs included in logic array 118.

Memory component or associated interconnection, configuration, or initialization information are down-load to program or configure reconfigurable logic devices or FPGAs included in logic array 118, reconfigurable memory cells 204, reconfigurable read/write I/O ports 206 or reconfigurable bidirectional latches 202 in memory array 200, or reconfigurable bidirectional switches or latches 112, 114, 116 in memory module 100.

Controller 102 also receives, stores and applies test or boundary scan vectors or patterns for verifying circuit functionality throughout memory module 100, preferably in accordance with the JTAG standard, as defined in IEEE Std. 1149.1: "Standard Test Access Port and Boundary Scan Architecture." Controller 102 is a conventional microprocessor coupled to time slot generator 104, MUX 110, and decoder 108 and receives instructions from a host

processor, which may be either user station 10 or debug unit 12.

Controller 102 uses a pre-specified instruction set provided in local disk or memory storage for processing the down-loading of partitioned netlist information into configurable memory circuits in memory array 200, preferably in accordance with the provided EDA tools.

Additional instructions are specified therein for configuring memory circuits in memory array 200 according to memory components provided in the netlist, in particular according to pre-defined configurations values such as port 206 access direction, digital storage content, storage depth size, storage width size and storage bank selections.

Decoder 108 is a memory configuration decoder circuit, embodied preferably as a programmable logic array (PLA) having 93 inputs, wherein 8 input pins are used for configuration code, 12 input pins are used for bank and sub-bank select, 24 input pins are used for read/write port select, 16 input pins are used for memory cell grouping, 8 input pins are used for memory cell enable, 24 pins are used for port time slot clock, and one input pin is used for system clock; and 136 outputs, wherein 32 output pins are used for write enable, 8 output pins are used for global bus enable, and 96 output pins are used for latch control.

Time slot generator 104 generates preferably up to 24 time slots or timed signal pulses for 24 ports. Each time slot has a 40-ns duration, wherein 10 ns is allocated for address time and 25 ns is allocated for access time. A programmable delay line may be applied before each time slot, and actual delay time depends on address to system clock delay.

Time slot generator 104 operates preferably under the control of controller 102 to provide tap output signals for enabling port read/write select and bank/sub-bank select to generate proper write signal and port data latch control. Additionally, time slot generator 104 adds read data delay time to system cycle time whenever the last or most recently read port is followed by more than 4 ports. Preferably, 150 ns of delay time is thereby added.

Logic array 118, which may be coupled to logic modules 14, includes reconfigurable logic circuits or FPGAs coupled to MUX 110 and switch or latch 12, 116, 114. Switch 112, latch 114 and latch 116 are reprogrammable bidirectional digital switches or latches, such as 74F543, which are each coupled to access ports 206 for accessing digital information for reading or writing to cells 204.

As shown in FIG. 3, MUX 110 is coupled to controller 102, decoder 108, memory array 200 and logic array 118; and

clock 106 is coupled to time slot generator 104 and decoder 108.

Preferably, memory module 100 is configured with predefined parameters. Memory depth is configured from 1 to 32,767. Bank width is limited to 32, 64, 128 or 256 bits. Thus, if a memory design uses 95 bits in width, then 128 bits are specified. Bank select configuration depend on bank width. For example, 128-bit bank allows no more than two banks because memory module 100 has a 256-bit width maximum.

Sub-bank select configuration is used for partial write operation. For example, 4-byte select is configured for 32-bit widths, 8-byte select is configured for 64-bit widths, 8 double-byte select is configured for 128-bit widths, and 8 quad-byte select is configured for 256-bit width. Sub-bank select may be disabled if partial write capability is not required. Moreover, one to 24 read ports and one to 16 write ports are preferably configured. The total number of read and write ports configured is 24.

Therefore, in accordance with the present invention, a memory circuit definition is implemented in at least one of various memory circuits 200, 208 having multiple access ports 206, wherein each access port 206 of each memory circuit 200, 208, in which the memory circuit definition is implemented, is configured for either reading from or writing to each such

memory circuit 200, 208.

Preferably, each memory circuit 200, 208 comprises static memory cells 204 having three bidirectional access ports 206, and the memory circuit definition defines initial storage contents, depth, width, and bank selection in the memory circuits 200, 208 according to predefined configuration values. Representative configuration values are provided in the attached Table I. Also the memory circuit definition may define, for each access port 206 of memory circuits 200, 208 having the implemented memory circuit definition, whether such access port 206 is configured for read or write mode.

Such configuration definition for memory access is achieved preferably by programming or configuring each bidirectional switch 112, latch 114, or latch 116, which is coupled to each of three access ports 206. In this way, each port 206 is implemented for write-only access (i.e., inbound signal direction) or read-only access (i.e., outbound signal direction). Each access port 206 may also be configured bidirectionally, for read and write access.

Optionally, the same memory circuit definition may be implemented in each of the memory circuits 200, 208, such that, for example, all access ports 206 for each memory circuit 200, 208 are configured identically.

Optionally, the configured access ports 206 may be accessed during predefined time slots provided by time slot generator 104 to read from or write to memory circuits 200, 208. Separate time slots may be predefined (e.g., prior to or during emulation) for reading from or writing to the memory circuits 200, 208, as opposed to non-memory logic circuits.

Time slots may be predefined as a function of external clock signals and memory circuit access times, and specified accordingly by the circuit designer, for example, during prototype circuit definition (netlisting) or verification (emulation). During a common or multiplexed time slot, a set of configured access ports 206 may be accessed in parallel; or alternatively, a single memory circuit 200, 208 may be accessed serially during each time slot (i.e., one at a time). Moreover, such accessing may be initiated by an asynchronous read or read-back trigger event.

Preferably, stored digital information is accessed from memory array 200 in words of equal to or smaller than a predefined size. Thus, when accessed words are larger than the predefined size, controller 102 partitions or parses such larger words into word portions which are each smaller than or equal to the predefined size. Additionally, each word portion is accessed preferably during different or subsequent time slots.

Optionally, a prototype circuit definition may be partitioned into a logic circuit definition for implementation in a reconfigurable logic module 14 and into the memory circuit definition for implementation in at least one of the memory circuits 200, 208. The reconfigurable logic module 14 may be interconnected directly, or through a programmable interconnect module 20, to the memory circuits 200, 208.

To emulate the partitioned prototype circuit definition, the reconfigurable logic module 14 and the memory circuits 200, 208 are interconnected and coupled to target system 18, which is then operated after so coupled. During emulation, the target system 18 may access the access ports 206 to read from or write to the memory circuits 200, 208. In a redefined configuration, target system 18 may also cause a different memory circuit definition to be implemented in the memory circuits 200, 208, whereupon the access ports 206 would be reconfigured accordingly.

In FIG. 4, a flow chart illustrates generally a method representative of the present invention. Initially, at least one memory circuit 100, 200, 208 is provided 210, wherein each memory circuit includes at least one static memory cell having preferably three access ports, wherein each port comprises a bidirectional switch. Then, a prototype circuit definition, preferably in the form of a netlist, is partitioned 220 into logic and memory portions.

In accordance with the present invention, the memory portion or definition is implemented 230 in at least one of the provided memory circuits, wherein such memory definition preferably defines intial configuration values. In addition, each access port 206 of the memory circuits having the implemented memory definition may be configured 240 for either reading or writing. Preferably during a predefined time slot, at least one of the configured access ports is accessed 250 to read or write from such memory circuits.

In the foregoing arrangement, prototype circuit emulation 260 is thereby enabled, and, as determined accordingly by the circuit designer, prototype circuit definition 220 and subsequent steps 230-260 may be repeated 270.

TABLE I

Configurations for 32-bit data word with 8-bit write select:

CELL:	PORT:	BANK:	SUB-BANK:	WRITE WIDTH:	READ WIDTH
8	16W 8R	8	4*8	32	32
	16W 4R	8	4	32	64
	16W 2R	8	4	32	128
	16W 1R	8	4	32	256
	8W 16R	8	4	32	32
	8W 8R	8	4	32	64
	8W 4R	8	4	32	128
	8W 2R	8	4	32	256
	8W 1R	8	4	32	256
4	8W 4R	4	4	32	32
	8W 2R	4	4	32	64
	8W 1R	4	4	32	128
	4W 8R	4	4	32	32
	4W 4R	4	4	32	64
	4W 2R	4	4	32	128
	4W 1R	4	4	32	128
2	4W 2R	2	4	32	32
	4W 1R	2	4	32	64
	2W 4R	2	4	32	32
	2W 2R	2	4	32	64
	2W 1R	2	4	32	64
1	2W 1R	1	4	32	32
	1W 2R	1	4	32	32
	1W 1R	1	4	32	32

Configurations for 64-bit data word with 8-bit write select:

CELL:	PORT:	BANK:	SUB-BANK:	WRITE WIDTH:	READ WIDTH
8	8W 4R	4	8*8	64	64
	8W 2R	4	8	64	128
	8W 1R	4	8	64	256
	4W 8R	4	8	64	64
	4W 4W	4	8	64	128
	4W 2R	4	8	64	256
	4W 1R	4	8	64	256
	2W 8R	4	8	64	64
	2W 4R	4	8	64	128
	2W 2R	4	8	64	256
	2W 1R	4	8	64	256
4	4W 2R	2	8	64	64
	4W 1R	2	8	64	64
	2W 4R	2	8	64	64
	2W 2R	2	8	64	128
	2W 1R	2	8	64	128
2	2W 1R	1	8	64	64
	1W 2R	1	8	64	64
	1W 1R	1	8	64	64

Configurations for 128-bit data word with 16-bit write select:

CELL:	PORT:	BANK:	SUB-BANK:	WRITE WIDTH:	READ WIDTH
8	4W 2R	2	8*16	128	128
	4W 1R	2	8	128	128
	2W 4R	2	8	128	128
	2W 2R	2	8	128	256
	2W 1R	2	8	128	256
4	2W 1R	1	8	128	128
	1W 2R	1	8	128	128
	1W 1R	1	8	128	128

Configurations for 256-bit data word with 32-bit write select:

CELL:	PORT:	BANK:	SUB-BANK:	WRITE WIDTH:	READ WIDTH
8	2W 1R	1	8*32	256	256
	1W 2R	1	8	256	256
	1W 1R	1	8	256	256

C L A I M S

1. A method for configuring a plurality of memory circuits, the method comprising the steps of:

providing a plurality of memory circuits, each memory circuit having a plurality of access ports;

implementing a memory circuit definition in at least one of the provided memory circuits; and

configuring each of the access ports of each provided memory circuit having the implemented memory circuit definition for either reading from or writing to each such provided memory circuit.

2. The method of claim 1, wherein each memory circuit comprises at least one static memory cell.

3. The method of claim 1 or 2, wherein each access port comprises a bidirectional switch configured according to the memory circuit definition.

4. The method of claim 1, 2 or 3, wherein only three access ports are provided to each memory circuit.

5. The method of claim 1, 2, 3 or 4, wherein the memory circuit definition defines initial storage contents in at least one of the provided memory circuits.

6. The method of any one of the preceding claims, wherein the memory circuit definition defines storage depth and width in at least one of the provided memory circuits.

7. The method of any one of the preceding claims, wherein the memory circuit definition defines storage bank selections in at least one of the provided memory circuits.

8. The method of any one of the preceding claims, wherein the memory circuit definition defines storage configuration in at least one of the provided memory circuits according to one of a plurality of predefined configuration values.

9. The method of any one of the preceding claims, wherein the memory circuit definition defines, for each of the access ports of each provided memory circuit having the implemented memory circuit definition, whether each such access port is configured for either reading from or writing to each such provided memory.

10. The method of any one of the preceding claims, wherein the same memory circuit definition is implemented in each of the provided memory circuits.

11. The method of any one of the preceding claims, further comprising the steps of accessing during a predefined time slot at least one of the configured access ports to read from or write to at least one of the provided memory circuits.

12. The method of claim 11, wherein a separate time slot is predefined for reading from and writing to the provided memory circuits.

13. The method of claim 11 or 12, wherein the time slot is predefined as a function of an external clock signal.

14. The method of claim 11, 12 or 13, wherein the time slot is predefined as a function of an access time of the provided memory circuits.

15. The method of any one of claims 11 to 14, wherein a plurality of the configured access ports associated with a plurality of provided memory circuits are accessed in parallel during a common time slot.

16. The method of any one of claims 11 to 14, wherein a single provided memory circuit is accessed during each predefined time slot.

17. The method of any one of claims 11 to 16, wherein accessing is initiated by an asynchronous read event.

18. The method of any one of claims 11 to 17, wherein words having a predefined size are accessed from each configured access port.

19. The method of claim 18, wherein each accessed word which is larger than the predefined size is partitioned into a plurality of word portions which are each no larger than the predefined size, and each word portion is accessed during different time slots.

20. The method of any one of the preceding claims, further comprising the step of:

partitioning a prototype circuit definition into a logic circuit definition for implementation in a logic circuit and into the memory circuit definition for implementation in at least one of the provided memory circuits.

21. The method of claim 20, wherein the logic circuit comprises a reconfigurable logic module interconnectable to at least one of the provided memory circuits.

22. The method of claim 21, wherein the reconfigurable logic module is interconnected to at least one of the provided memory circuits through a programmable interconnect module.

23. The method of claim 20, 21 or 22, further comprising the step of:

emulating the partitioned prototype circuit definition by coupling the logic circuit and at least one of the provided memory circuits to a target system and operating the coupled target system.

24. The method of claim 23, wherein the coupled target system accesses at least one of the configured access ports to read from or write to at least one of the provided memory circuits.

25. The method of claim 23, wherein the coupled target system causes a different memory circuit definition to be implemented in at least one of the provided memory circuits, and also causes each of the access ports of each provided memory circuit having the implemented different memory circuit definition to be reconfigured.

26. The method of any one of claims 1 to 9, further comprising the steps of:

implementing a different memory circuit definition in at least one of the provided memory circuits; and

reconfiguring each of the access ports of each provided memory circuit having the implemented different memory circuit definition for either reading from or writing to each such provided memory circuit.

27. Apparatus for configuring a plurality of memory circuits, the apparatus comprising:

a plurality of memory circuits, each memory circuit having a plurality of access ports; and

a prototype circuit definition partitionable into a logic circuit definition for implementation in a logic circuit and into a memory circuit definition;

wherein the memory circuit definition is implementable in at least one of the memory circuits, each access port of each memory circuit having an implemented memory circuit definition is configurable for either reading from or writing to each such memory circuit, and a configured access port reads from or writes to at least one of the memory circuits during a predefined time slot.

28. Apparatus for configuring a plurality of memory circuits, the apparatus comprising:

a plurality of memory circuits, each memory circuit having a plurality of access ports; and

a memory circuit definition implemented in at least one of the memory circuits, wherein each of the access ports of each memory circuit, in which the memory circuit definition is implemented, is configured for either reading from or writing to each such memory circuit.

29. A method for configuring a plurality of memory circuit, the method being substantially as hereinbefore described with reference to and as illustrated in the accompanying drawings.

30. Apparatus for configuring a plurality of memory circuits, constructed and arranged to operate substantially as hereinbefore described with reference to and as illustrated in the accompanying drawings.

Patents Act 1977 Examiner's report to the Comptroller under Section 17 (The Search report) -29-	Application number GB 9411924.5
Relevant Technical Fields (i) UK Cl (Ed.M) G4C C700B, C800M; G4A (AMX) (ii) Int Cl (Ed.5) G06F 12/06; G11C 7/00, 8/00	Search Examiner B J EDE Date of completion of Search 21 SEPTEMBER 1994
Databases (see below) (i) UK Patent Office collections of GB, EP, WO and US patent specifications. (ii) ONLINE DATABASES: WPI	Documents considered relevant following a search in respect of Claims :- 1-30

Categories of documents

X:	Document indicating lack of novelty or of inventive step.	P:	Document published on or after the declared priority date but before the filing date of the present application.
Y:	Document indicating lack of inventive step if combined with one or more other documents of the same category.	E:	Patent document published on or after, but with priority date earlier than, the filing date of the present application.
A:	Document indicating technological background and/or state of the art.	&:	Member of the same patent family; corresponding document.

Category	Identity of document and relevant passages	Relevant to claim(s)
X	EP 0149451 A2 (WANG) whole document relevant	1,27 and 28 at least

Databases: The UK Patent Office database comprises classified collections of GB, EP, WO and US patent specifications as outlined periodically in the Official Journal (Patents). The on-line databases considered for search are also listed periodically in the Official Journal (Patents).